iLab-20M: A large-scale controlled object dataset to investigate deep learning

Ali Borji[1] Saeed Izadi[2] Laurent Itti[3]
[1]Center for Research in Computer Vision, University of Central Florida
[2]Amirkabir University of Technology, [3]University of Southern California
aborji@crcv.ucf.edu, sizadi@aut.ac.ir, itti@usc.edu

Abstract

Tolerance to image variations (e.g., translation, scale, pose, illumination, background) is an important desired property of any object recognition system, be it human or machine. Moving towards increasingly bigger datasets has been trending in computer vision especially with the emergence of highly popular deep learning models. While being very useful for learning invariance to object inter- and intra-class shape variability, these large-scale wild datasets are not very useful for learning invariance to other parameters urging researchers to resort to other tricks for training models. In this work, we introduce a large-scale synthetic dataset, which is freely and publicly available, and use it to answer several fundamental questions regarding selectivity and invariance properties of convolutional neural networks. Our dataset contains two parts: a) objects shot on a turntable: 15 categories, 8 rotation angles, 11 cameras on a semi-circular arch, 5 lighting conditions, 3 focus levels, variety of backgrounds (23.4 per instance) generating 1320 images per instance (about 22 million images in total), and b) scenes: in which a robotic arm takes pictures of objects on a 1:160 scale scene. We study: 1) invariance and selectivity of different CNN layers, 2) knowledge transfer from one object category to another, 3) systematic or random sampling of images to build a train set, 4) domain adaptation from synthetic to natural scenes, and 5) order of knowledge delivery to CNNs. We also discuss how our analyses can lead the field to develop more efficient deep learning methods.

1. Introduction

Object and scene recognition is arguably the most important problem in computer vision and while humans do it quickly and almost effortlessly, machines still lag behind humans. In some cases, where variability is relatively low (e.g., frontal face recognition) machines outperform humans but they do not perform quite as well when variety is high. Hence, the crux of the object recognition problem is tolerance to intra- and inter-class variability, lighting, scale, in-plane and in-depth rotation, background clutter, etc [9].

Thanks to big data and deep neural networks, computer vision has recently enjoyed a rapid progress, witnessed by high accuracies over the ImageNet dataset (top-5 error rate between 3-10% over 1,000 object categories). Recent models (e.g., Alexnet [31], VGG [54], Overfeat [50], GoogLeNet [57], and ResNet [23]) have surpassed previous scores in several benchmarks such as generic object and scene recognition [31, 54], object detection [50, 20], semantic scene segmentation [6, 20], face detection and recognition [66], texture recognition [7], fine-grained recognition [39], multi-view 3D shape recognition [56], activity recognition [53, 28], and saliency prediction [32].

One chief concern regarding the wild large-scale benchmarks and datasets, however, is the lack of control over data collection procedures and deep comprehension of stimulus variety. While existing large-scale datasets are very rich in terms of inter- and intra-class variability, they fail to probe the ability of a model to solve the general invariance problem. In order words, natural image datasets (e.g., ImageNet [8], SUN [64], PASCAL VOC [14], LabelMe [48], Tiny [61], and MS COCO [38]) are inherently biased in the sense that they do not offer all object variations [60]. To remedy this, some works (e.g., [45, 35, 41]) have resorted to synthetic datasets where several object parameters exist.

Ideally, we would like models to be tolerant to identity-preserving image variations (e.g., variation in position, scale, pose, illumination, occlusion). To probe this, some researchers have used synthetic home-brewed datasets either by taking pictures of objects on a turntable (e.g., NORB [35], COIL [41], SOIL-47 [29], ALOI [19], GRAZ [42], BigBIRD [55]) or by constructing 3D graphic models and rendering textures to them (e.g., Pinto *et al.*[45], Peng *et al.* [43]). While proven to be beneficial in the past, these datasets are very small for training deep neural networks with millions of parameters. Further, they usually have small number of classes, instances per class, background variability, in plane and in-depth rotation, illuminations, scale, and total number of images. Here, to remedy these shortcomings, we introduce a large-scale controlled object dataset with rich variety and a larger set of images.

Dataset	Ref	Domain	Object Classes	Objects per Class	Backgrd per obj	Views per obj+bg	Bounding Box?	Object Contours?	Total Images
COIL	[41]	Handheld	100	1	1	72	Implicit	No	7,200
SOIL-47	[29]	Handheld	—	47	1	42	Implicit	No	1,974
Pascal	[14]	Misc	20	790-10,129	1	1	Yes	Partial	11,540
Caltech-101	[15]	Google	102	31-800 ($\mu = 90$)	1	1	No	No	9,144
Caltech-256	[22]	Google	257	80-827 ($\mu = 119$)	1	1	No	No	30,607
LabelMe	[48]	Misc	900	?	$\sim$1	$\sim$1	Partial	Partial	62,197 (a)
NORB	[35]	Toys	5	10	1 (b)	1,944	Implicit	No	48,600 (b)
FERET	[44]	Faces	1	1,199	1	1-24	Yes	No	14,051
MNIST	[34]	Digits	10	6,000	1	1	Implicit	No	60,000
ETHZ	[17]	Natural	5	32-87	1	1	Yes	Yes	255 (c)
TINY	[61]	Web	75,062	?	1 (?)	1	Implicit	No	79,302,017 (d)
CIFAR-100	[30]	Web	100	600	1	1	Implicit	No	60,000 (d)
ALOI	[19]	Handheld	1,000 (e)	$\sim$1	1	108	Implicit	No	110,250
GRAZ	[42]	Photographs	4	311-420	1	1	No	Partial	1,476
CoPhIR	[3]	Flickr	? (f)	?	1 (?)	1 (?)	No	No (f)	106,000,000
ImageNet	[8]	Misc	21,841	$\sim$1	$\sim$1	$\sim$1	Yes	No	14,197,122
SUN	[64]	Misc	3,819	(g)	1	1	Yes	Yes	131,067
MS COCO	[38]	Misc	91	$\sim$5,000	1	1	Yes	Yes	328,000 (a)
RGB-D	[33]	Household	51	$\sim$6	1	250	Yes	No	250,000
Big-BIRD	[55]	Household	100	1	1	600	Yes	No	250,000
iLab-20M	–	Toy vehicles	15	25-160	14-40	1,320	Implicit	No	21,798,480

Table 1. Overview of some popular object recognition datasets. The last one proposed here avoids the dreaded entry of "1" in any column of the table. Implicit bounding box means that it can be trivially computed (e.g., objects are centered within images). Notes: (a) Still growing. (b) Many additional images were created by digitally jittering objects and compositing various backgrounds. (c) 289 objects in 255 images. (d) Image resolution 32×32. Note that CIFAR is a subset of the TINY dataset. (e) 1,000 objects total, not grouped by categories. (f) MPEG-7 and Flickr user tags (e.g., summer, Paris, China) available. (g) The number of instances per object category shows the long tail phenomenon: a few categories have a large number of instances (window: 16,080, chair: 7,971, wall: 20,213) while a majority of them have a relatively modest number of instances (airplane: 179, floor lamp: 276, boat: 349).

2. Related work

Several controlled datasets have been introduced in the past which have dramatically helped progress in computer vision (Table 1). Two famous examples are FERET face [44] and MNIST digit [34] datasets. Nowadays, we have face and digit recognition systems that perform either at the level of humans (e.g., [58]) or superior (perhaps not as robust due to variations and noise). Similar datasets are available for generic object recognition but lack characteristics of a large-scale representative dataset covering many sorts of invariance (e.g., background clutter, shape, occlusion, size). For example, the COIL dataset [41], which also used a turntable to film 100 objects under various lightings and poses, contains one object instance per category (e.g., one telephone, one mug). The larger ALOI dataset [19] contains 1,000 objects but few instances per category. The NORB dataset [35] has 50 small toy objects (10 instances in each of 5 categories). Almost all available turntable datasets are small scale and not very rich in terms of variations.

Previous research using controlled datasets, such as turntables images, has been mainly focused on inspecting models or brewing concepts and ideas. Some recent works have attempted to show that there is a real benefit of these datasets in transferring knowledge to large-scale natural scene datasets [26, 67]. This has been studied under the names of *domain adaptation*, *task transfer*, or *multi-task learning*. The idea here is that knowledge gained from a controlled dataset (or task), via turntables or graphic models, can be transferred to real-world naturalistic datasets with even different statistics (e.g., texture). For example, Peng *et al.* [43] trained models on an augment of synthetically generated images (using a 3D graphics object model) and natural scenes (from ImageNet and PASCAL) and reported an improvement in accuracy over the latter datasets. They, however, did not probe whether the improvement was due to learning better invariance or instance level variety and richness. Some other works have also advocated similar directions [21, 49, 11, 16].

Another motivation for utilizing controlled datasets comes from neuroscience and cognitive vision literature. CNNs were initially inspired by the hierarchical structure of the visual ventral stream [18]. They were later used to explain some physiological and behavioral data of humans and monkeys (e.g., [46, 52, 65, 51]). It has been asserted that humans learn invariance with few exemplars a.k.a. zero- or one-shot learning. This is the opposite of the way that CNNs currently learn recognition. These models need an enormous amount of labeled data. In this work, we explore the ways a large-scale controlled dataset, containing rich information regarding various object parameters, can be utilized to improve object recognition performance. It is important to be aware of human performance to gauge the progress [4]. Just recently, He et. al. [24] reported a top-5 error rate of 4.9% on ImageNet which is lower than 5.1% human error rate on this dataset [47]. This raises some questions: Have models surpassed humans? If yes, in what aspects? Is it theoretically possible to achieve a better performance than humans on these problems? etc.

Another related area to our work, which naturally fits well to turntable setups, is the manifold embedding and di-

mensionality reduction. These techniques try to preserve and leverage the underlying low dimensional manifold in data in supervised or unsupervised manners (e.g., [69, 59]). For instance, Weston *et al.* [63] introduced an embedding-based regularizer to impose the same labels for the neighboring training samples to benefit from the structure in the data. They used gradient descent to optimize the regularizer and adopted it for CNNs. Another classic example is Siamese Networks [5] which are two identical copies of the same network, with the same weights, fed into a 'distance measuring' layer to compute whether the two examples are similar or not. Given the labeled data, the network encourages similar examples to be close, and dissimilar ones to have a certain minimum distance from each other. While these techniques have been applied to controlled datasets, their usefulness over large-scale controlled datasets still remains to be explored. Our proposed dataset can be helpful in this direction as it combines the best of the two worlds: *instance-level variety* of large-scale datasets and *rich parametrization* of controlled synthetic images which are precious to study probing the behavior of CNNs.

3. The iLab-20M dataset

Many image datasets have been proposed to assist machine vision algorithm development and testing (Table 1). Those datasets which have provided large collections of training exemplars per well-defined object category have been useful in advancing the state of the art. Excellent examples include FERET for face recognition [44], with 14,051 images of 1,199 individuals in one class (human faces), or MNIST for handwritten digits [34], with 60,000 images in 10 classes from 500 writers. Today, recognizing faces or handwritten digits is considered a reasonably well solved problem, although of course improving tolerance to noise and other nuisance parameters is always possible.

In other domains, including recognition of objects from generic categories, most efforts have focused on providing very useful test sets and performance challenges (e.g., ImageNet [8]), but these often lack in the sheer volume of training exemplars provided within each object category and for each object instance, lack pose information, and often contain occlusions. This limits their usefulness for training. For example, the 'calculator' category of Caltech-256 [22] contains 100 images of what appears to be 100 different calculators with no pose data. While this is highly appropriate for testing, we hypothesize that training can be greatly improved by using many different views of different instances of objects in a number of categories, shot in many different environments, and with pose information explicitly known. Indeed, biological systems can rely on object persistence and active vision to obtain many different views of a new physical object. In humans and monkeys, this is believed to be exploited by the neural representations [37], though

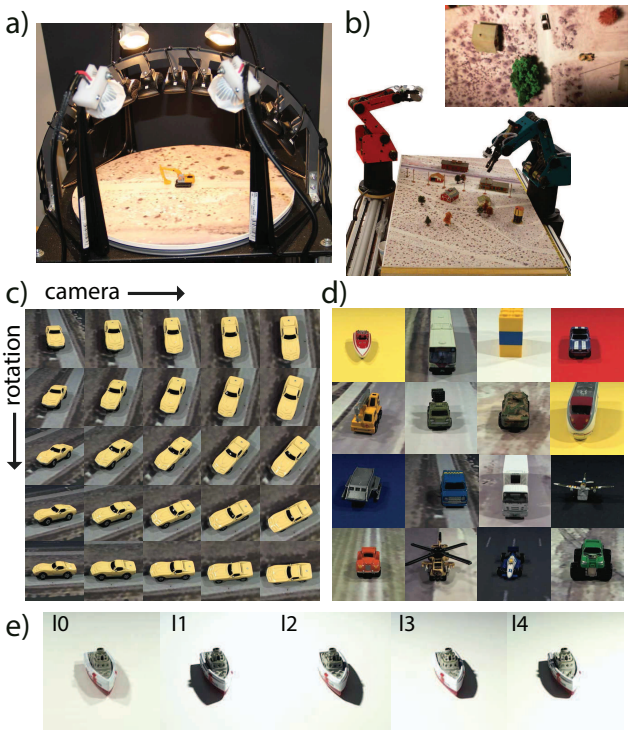

Figure 1. Turn-table photo shooting setup. a) turntable with 8 rotation angles, 11 cameras on a semicircular arch, 4 lighting sources (generating 5 lighting conditions), 3 focus values and random backgrounds (overall $8 \times 11 \times 5 \times 3 = 1320$ images for each instance per background). Recording parameters are: resolution 960×720, color mode YUYV, brightness 128, contrast 32, saturation 32, gain 30, auto white balance off, manual white balance temperature 3100K, sharpness 72, auto exposure off, auto focus off, focus base value 97-119. b) robotic-assisted arms, one holding the camera, the other taking wide-field pictures from random viewpoints and distances. c) a sample instance of a car from 5 consecutive rotations and 5 consecutive arch cameras. d) a sample instance from each object category (same lighting, rotation and focus; all set to zero) presented in the order shown in Table 2. e) an instance of a boat under different illuminations.

the exact mechanisms remain poorly understood. Although adult humans can learn new object instances from a single view, one should not forget that this ability might only emerge at the culmination of a long evolutionary process plus 20-some years of individual training.

Popular datasets fall short in at least one dimension, be it the number of classes, objects per class, number of backgrounds/environments, or views per object, as shown in Table 1. Particularly relevant to our effort are: 1) COIL [41], which also used a turntable to film 100 objects under various lighting and poses; however, COIL only contains one object instance per category and only black backgrounds (similar to the larger ALOI dataset with 1,000 objects and a few per category [19]), and 2) NORB [35] with 50 small toy objects similar to the ones we used (10 instances in each of 5 categories); however, all objects were painted uniformly and shot in grayscale on blank backgrounds (different backgrounds were later composited digitally).

Parameter \ Category	Boat	Bus	Calib-ration	Car	Equip-ment	Military	Tank	Train	UFO	Van	Semi Truck	Plane	Pickup Truck	Heli-copter	F1-car	Monster Truck
Num objects	27	25	13	160	64	54	31	25	40	29	33	85	40	25	40	40
Num bg (mean)	20	21.3	1	26.1	21.6	18.5	30.3	37	29	29.4	23.1	18.4	30.1	23.2	14	21.5
Num bg (std)	0.0	1.5	0.0	1.3	1.3	0.9	7.8	0.0	4.4	0.9	5.0	3.3	4.9	10.6	0.0	4.8
Num bg (min-max)	20-20	20-23	1-1	24-28	20-23	18-20	20-36	37-37	26-37	28-30	17-27	17-26	25-35	14-35	14-14	14-25
Total images (K)	713	704	17	5518	1822	2611	1432	462	739	933	1113	1907	1505	660	950	1426
Size (GB)	551	545	11	4300	1500	2100	1200	363	565	724	874	1400	1200	495	722	1100
Used here	✓	✓	-	-	-	-	✓	✓	✓	✓	-	-	-	-	✓	-

Table 2. Summary statistics of iLab-20M dataset. There are 21,798,480 images in total from 16 categories (one used for calibration purposes only) with 25 to 160 instances per category. Five parameters include: 11 cameras on an arch, 4 lighting sources on 4 corners (5 conditions), 8 horizontal rotations, 132 backgrounds (7 solid color) and 3 focus values. Average number of backgrounds per object instance is 23.39. There are 46 unique backgrounds per category (average backgrounds per object 145.76 with std = 162.62; min = 25, max = 731). Total size of the dataset with resolution 960 × 720 is 17.65TB. The cropped version of the images (256 × 256 pixels) is also available with 2.2TB in size. Total number of images per category is rounded to save space.

3.1. Turntable setup

The turntable consists of a 14"-diameter circular plate actuated by a robotic servo mechanism. A CNC-machined semi-circular arch (radius 8.5") holds eleven Logitech C910 USB webcams which capture color images of the objects placed on the turntable (Fig. 1.a). A micro-controller system actuates the rotation servo mechanism and switches on and off four LED lightbulbs (Ecosmart ECS 16 WW FL, 295 lumens each, color rendering index 87, correlated color temperature 3000K). Lights are controlled independently, in 5 conditions: all lights on, or one of the four lights on.

Cameras were connected to a Linux computer (6-core AMD Phenom CPU, 16GB RAM) with 11 independent USB controllers. Camera settings were as follows, using the Linux V4L2 driver: resolution 960 × 720, color mode YUYV, brightness 128 (default for these cameras), contrast 32 (default), saturation 32 (default), gain 30, auto white balance off, manual white balance temperature 3100K, sharpness 72 (default), auto exposure off, manual exposure 125 (all lights on) or 450 (one light on), autofocus off, focus base value 97 - 119 depending on the camera. Objects were mainly Micro Machines toys (Galoob Corp.) and N-scale model train toys, as shown Fig. 1.d. These objects present the advantage of small scale, yet demonstrate a high level of detail and, most remarkably, a wide range of shapes (i.e., many different molds were used to create the objects, as opposed to just a few molds and many different painting schemes). Backgrounds were 125 color printouts of satellite imagery from the Internet, and 7 plain solid-color backgrounds (white, red, blue, yellow, etc). Every object was shot on all solid-color backgrounds, for possible later compositing of additional digital backgrounds, and for possible reconstruction of 3D models. Every object was shot on at least 14 backgrounds, in a relevant context (e.g., cars on roads, trains on railtracks, boats on water).

In total, 1,320 images were captured for each object and background combination: 11 azimuth angles (from the 11 cameras), 8 turntable rotation angles, 5 lighting conditions, and 3 focus values (-3, 0, and +3 from the default focus of each camera). Each image was saved with lossless PNG compression (∼1 MB per image). The complete dataset hence consists of 704 objects, each shot on 14 or more backgrounds, with 1,320 images per object/background combination, or almost 22M images (See Table 2). The dataset is freely available and distributed on 3 8TB hard drives.

3.2. Robotics-assisted model scenes

In addition, we created robotics-assisted model scenes to record broader scenes where objects were placed in variable contexts. The long-term motivation for this larger scenery is to collect many images which can be used to test algorithms both on their ability to first locate and then to recognize objects, and on their possible ability to exploit larger scene contexts to aid recognition (see, for example [12, 25]).

The robotics-assisted scenes (Fig. 1.b) consist of a 40"×29" table onto which 1:160 poster prints of satellite images (e.g., Google maps) are placed (corresponding to a real-world area of 195m×118m). One 8-axis robot arm holds a camera (Microsoft LifeCam Cinema, 1280×720, YUYV) which can be placed and oriented at any location and pose reachable by the arm. A second arm holds a light source (Jingsam LED 7W, 437 lumens, 3000K).

The robots are programmed in two ways: 1) pseudo-random motion, generating flybys, 2) point to specific locations on the table and shoot objects from different viewpoints and distances. An interactive user interface assists in configuring a scene for robotics-assisted filming.

4. Experiments and results

To start exercising the dataset, we tested it on small subsets of the available data. To understand generalization across image variations (object shape, object viewpoint, lighting, etc), CNNs are evaluated by taking slices of the dataset. We utilize pre-trained Alexnet [31] (on ImageNet) and fine-tune it on iLab-20M. The behavior of off-the-shelf features is investigated in our analyses as well. We use 7 object categories (out of 16) and avoid data augmentation as we have flipped versions of the objects from the turntable. The label layer contains several units depending on the task

(2, 4 or 7 for object categorization; variable number of units for parameter prediction). We report average accuracies and standard deviations where there is randomness in the experimental procedure. Experiments are performed using the publicly available Caffe toolkit [27] ran on a Nvidia Titan X GPU and Ubuntu 14.04 OS.

We aim to answer these questions: Can a pre-trained CNN model predict the setting parameters such as lighting source, degree of azimuthal rotation, degree of camera elevation, etc? Can it transfer the learned knowledge from one object category to another? Which parameters are more important in the transfer? How much knowledge can a model transfer from iLab-20M to the ImageNet? Which one is a better strategy to make an object dataset: random or systematic image harvesting? How the order of learning parameter invariance influences overall network parameter tolerance and accuracy? Some of these questions have been addressed in the past to some extent [1, 68, 70, 10, 40].

4.1. Selectivity and invariance

Humans are very good at predicting the category of an object and also telling about its parameters. Human visual system is selective to object category and invariant to parameters and variations. In this experiment, we aim to systematically investigate this competition for two layers of the Alexnet: *pool5* and *fc7*. We probe the expressive power of these layers for object category and parameter prediction.

Four categories from iLab-20M (out of 16) were chosen for this analysis including *boat, bus, tank* and *ufo*. Images were lumped to train a SVM classifier. All features were normalized to have zero-mean before feeding to the classifier. The dimensionality was reduced to N-dimensions using SVD, where N refers to the number of instances in the training set. The reported results are average accuracy over random 5-fold cross validation test sets, each of size 2K. We trained two SVMs, one for category prediction and another for parameter prediction. Results are shown in Fig. 2.

As expected, we see that *fc7* features result in a high classification accuracy, however, the surprising salient result is the shoulder-to-shoulder performance of *pool5* and *fc7* layers. Relying on this outcome, it seems that both *fc7* and *pool5* representations convey useful discriminative information for object recognition. Comparing the performance

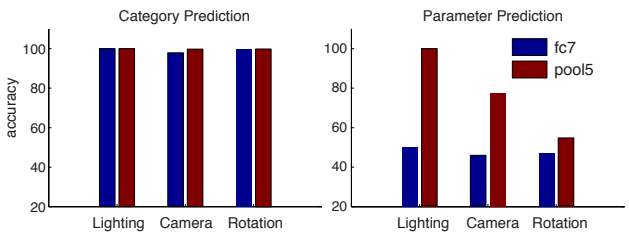

Figure 2. Selectivity and invariance. Expressive power of Alexnet *pool5* and *fc7* layers for category and parameter prediction on a 4 class problem.

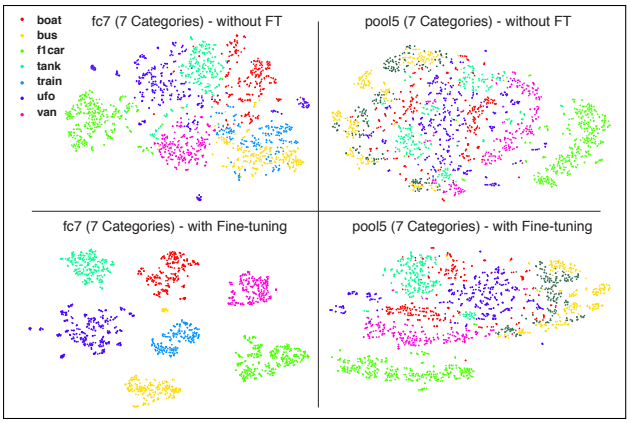

Figure 3. t-SNE representation of the Alexnet layers. The *fc7* representation works remarkably well at recognizing objects as they are mutually linearly separable after fine-tuning. Further, *pool5* representation does not contain discriminative information compared to *fc7*. This figure also demonstrates the effect of fine-tuning. Distribution of samples for different categories tend to become very compact after fine-tuning. Fine-tuning does not seem to add more discriminative power to the *pool5* representation.

over parameter prediction, one can notice the superiority of *pool5* layer over *fc7*. This is consistent with the work by Bakry *et al.* [2] where they analytically found that fully connected layers make effort to collapse the low-dimensional intrinsic parameter manifolds to achieve invariant representations. However, only view manifold was taken into consideration in Bakry *et al.*'s work, while here we analyze the behavior of more common parameters.

In brief, our results suggest that the feature space spanned by *pool5* layer contains more information than *fc7* layer for parameter prediction. At the same time, the very representation forces different categories to be highly apart from each other (thus keeping the structure of manifolds as linearly-separable as possible for different categories). The representation by *fc7* sensibly discards parameter information to become invariant while keeping the categories as separable as possible. We observe that the layer just before fully connected layers provides better compromise between categorization and parameter estimation.

Parameter prediction accuracies for lighting (5 classes), turntable rotation (4 classes), and camera view (6 classes) in order are 100%, 62%, and 77%. These figures suggest that camera view (considering the normalized-to-chance accuracy) has the most complex structure for parameter prediction whereas the lighting is simpler. This is somewhat sensible since changing camera view leads to geometric shape variations, and ports the prediction task into a much more difficult problem to address. In contrast, lighting variations do not alter the shape of the object, and are thus easier to capture. Note that this result is on our data and may not necessarily scale to natural scenes.

We use the t-SNE dimensionality reduction method in [62] to visualize the learned representations over seven

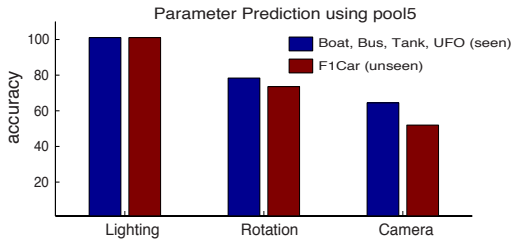

Figure 4. Knowledge transfer over object categories with one parameter changing. Alexnet is trained over four object classes and is tested on the same or different object classes (over different instances).

categories of iLab-20M along with variation parameters (See Figure 3). The *fc7* representation works remarkably well at recognizing objects as they are mutually linearly separable after fine-tuning. Further, *pool5* representation does not contain discriminative information compared to *fc7*. Please see also the supplement for more details.

4.2. Knowledge transfer

Humans are very efficient at estimating and transferring parameters of a seen object to another unseen object in complicated scenarios. For example, they can reliably estimate the lighting source direction of an object and tell whether another object has been subject to the same lighting exposure. Complementary to our previous analysis, in this experiment, we aim to assess the power of CNNs in transferring the learned parameter over one object category to another. We focus on *pool5* layer here since as we discussed above, *fc7* layer is invariant to parameters and is thus less useful for discriminating between different parameters.

All parameters were fixed except one (i.e., slicing the dataset along only one parameter). We included instances from four categories (*boat, bus, tank, ufo*) in the training set, and tested the learned knowledge on instances from an unseen category (*f1car*) as well as 4 seen categories (but different instances). We utilized the *pool5* representation and reduced the dimensionality to N, where N refers to number of samples. The 5-fold cross validation average accuracy for parameter prediction is shown in Fig. 4.

Results show a decent degree of knowledge transfer. As Fig. 4 exhibits, the lighting parameter is relatively easier to be transferred to unseen categories. It has a head-to-head accuracy across seen and unseen categories. On the other hand, knowledge transfer for rotation and camera view parameters is accompanied with sensible degradation in performance. In summary, we see that the knowledge is promisingly transferable across seen and unseen categories. The degradation in rotation and camera prediction is intuitively justifiable as these parameters are highly dependent on the 3D properties of the object shape (See also [36]).

4.3. Systematic and random sampling

Large-scale datasets have been so far constructed by harvesting images randomly from the web. The major reason-

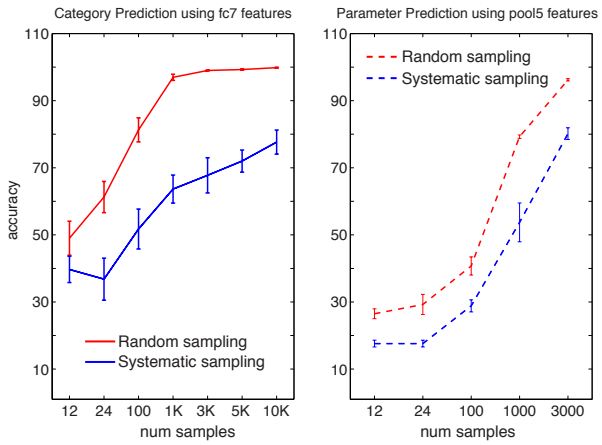

Figure 5. Analysis of two sampling strategies over a 4-class classification problem (boat, bus, tank, ufo). Left: category prediction accuracy using *fc7* features. Right: Parameter prediction accuracy using *pool5* features.

ing for doing so is to include as much variability (mainly intra- and inter-class variation) as possible in the dataset. It has not yet been systematically studied whether this is a good strategy compared to controlled strategies conducted in turntable setups. In this analysis, we consider two strategies to find the answer: *i*) *Random* strategy where n samples (across all parameters and instances) are chosen randomly and are used to train an SVM to predict the object category, and *ii*) *Systematic (or exhaustive)* strategy, in which an object instance is chosen randomly and then other images from that object are added to our training set, by scanning all parameters, until n samples are reached. We assume availability of a fixed limited budget (time or cost) enough for processing only n samples.

We addressed a 4 class problem (*boat, bus, tank, ufo*) by increasing n starting from 12 up to 10,000 samples. In each experiment, $n/4$ samples were chosen randomly from all 4 categories across all parameters, and were fed into the Alexnet to get the *fc7* (or *pool5*) representation. Then, we trained a linear SVM classifier on this data. A fixed test set of size 500 was randomly selected from all categories with all parameters and was kept fixed during the analysis. We measured category prediction at *fc7* and parameter prediction at *pool5*, reducing the dimensionality to 2,500 for all values of n in the latter. Results are shown in Fig. 5.

We observe that random sampling strategy performs better in category prediction. This makes sense since randomly choosing images offers more instance level variety (better than systematic) leading to better recognition. Interestingly, and counter-intuitively, we see that random strategy works better in parameter prediction as well. We believe that the parameter prediction is somewhat dependent on the 3D properties of object shape, and since in the systematic strategy, the learner is not faced with sufficient instances, it fails to predict parameters compared to random strategy. Overall, what we learn is that instance level variation is of

train \ test	Without fine tuning		With fine tuning	
	Natural	iLab-20M	Natural	iLab-20M
Natural	95	75	93 ↓	65 ↓
iLab-20M	78	97	70 ↓	100 ↑

Table 3. Domain adaptation on boat vs. tank classification (in percentage).

train \ test	Without fine tuning		With fine tuning	
	Natural	iLab-20M	Natural	iLab-20M
Natural [2000]	96.48 (0.5)	55.6 (2.7)	95.56 (0.6)	68.06 (2.0)
iLab-20M [2000]	66.92 (3.2)	96.90 (0.2)	65.22 (1.4)	99.72 (0.1)
iLab-20M [1000] + Natural [1000]	94.42 (0.8)	93.94 (0.4)	92.52 (0.2)	98.70 (0.2)

Table 4. Domain adaptation over a 4-class problem (boat, tank, bus, and train). Numbers in parentheses are standard deviations.

high importance for both category and parameter prediction and this is perhaps why the systematic sampling strategy is hindered. Thus, in dataset creation, it is vitally advantageous to have as much instance level variation as possible.

4.4. Domain adaptation

Currently, there is a gap in relating results learned over synthetic datasets to results learned on large-scale datasets. We train models on iLab-20M and apply them to natural scenes (and vice versa) to see how much knowledge they can transfer from one dataset (source domain) to another (target domain). This way, we can also discover along which dimension(s) a dataset varies the most and whether it offers sufficient variability for learning invariance. In other words, we can somehow indirectly measure dataset bias [60]. Ultimately, it is desirable to generalize what is learned from synthetic datasets to natural scene datasets.

We consider two scenarios: *i)* a binary classification problem *boat* vs. *tank*, and *ii)* a 4-class problem including *boat, tank, bus* and *train*. In each scenario, we train a SVM (using *fc7* representation) from either natural scenes (selected from ImageNet) or iLab-20M and apply it to the other dataset. We also merge images from the two datasets and measure the accuracy on each individual dataset. We consider both off-the-shelf features of the Alexnet (pre-trained over ImageNet) and fine-tuned features over iLab-20M.

Augmenting data along all parameters: Here, we choose images along all parameters. Results in Table 3 show that training on each type of image, expectedly works the best on the same type of test image (95% from ImageNet to ImageNet and 97% from iLab-20M to iLab-20M). Cross application of models results in lower (but above 50% chance) accuracy. We observe that fine tuning the Alexnet on iLab-20M boosts the performance on iLab-20M to 100% while hindering the accuracy over ImageNet as CNN features are now tailored (and are hence selective) to our images.

Table 4 shows domain adaptation results over 4 classes. Results align with accuracies over 2 classes, although accuracies are lower here. Here, again combining images from datasets hinders performance over each individual dataset due to contamination of features. The reason why performance is low when applying a model trained on iLab-20M to ImageNet is mainly because objects in these two datasets have different textures and statistics which demand more sophisticated ways of domain adaptation.

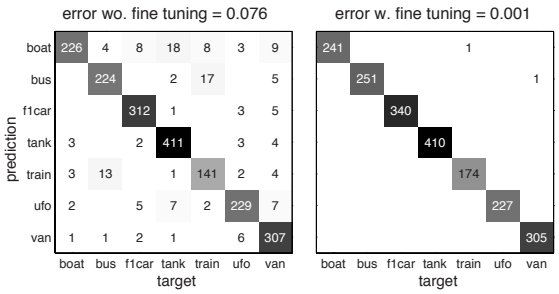

Figure 6. Confusion matrices of Alexnet over seven categories of the iLab-20M dataset without (left) and with fine tuning (right).

Accuracies over 2-class and 4-class problems are very high ($> 95\%$). To further investigate accuracy of Alexnet, we increased the number of classes to 7. As seen in the confusion matrices in Fig. 6, fine tuning the network increases the accuracy from 92.5% to 99.9% with only two mistakes[1].

Augmenting data along a single parameter: Here, we investigate which parameter is more effective in domain-adaptation (from synthetic to natural images.). Two categories, existing in both datasets, are considered: *boat* and *tank*. To form a training set, we vary only one parameter at a time while keeping all others fixed. Then, *fc7* features are computed for the training set and a linear SVM is trained. The same features are computed for natural images and the learned model on synthetic samples is tested on them. For each parameter, we had 275 synthetic images for training and a fixed set of 3,000 images from ImageNet for testing.

In a complementary experiment, all parameters were allowed to vary except one (opposite of the above). A set of 2,000 samples were randomly selected (complying with the conditions) and a linear SVM was trained on them (using fc7). The parameter whose absence drops the accuracy more is considered to be more dominant. 5-fold cross validation accuracies are reported in Fig. 7.

As shown in the bar chart in Fig. 7, the camera-view is of the highest importance as it leads to the highest accuracy on the fixed natural test set. This is reasonable since real world objects are often viewed from angles at different degrees of elevation (in-depth rotation). We thus speculate that camera-view might be the dominant varying parameter in natural scenes. The (in-plane) rotation is the next important parameter as it gains the next top accuracy on natural images. Surprisingly, the lighting source is ranked as the

[1]Please see the supplementary material for t-SNE visualization [62] of without- and with fine-tuned *fc7* and *pool5* features.

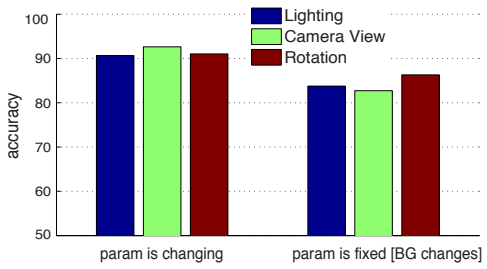

Figure 7. Domain adaptation with a single parameter change.

least effective parameter in our analysis. The absence of camera-view drops the recognition accuracy more than the other two parameters (the right side bars in Fig. 7).

4.5. Analysis of parameter learning order

In this analysis, we study whether/how the order of knowledge delivery to CNNs matters. First, we prepare two datasets (training with 40K images, validation with 10K) from four categories (*boat, bus, tank* and *train*) and annotate them with rotation labels. Alexnet is fine tuned on the training set. We set the learning rate for all the layers to 0.001, except *fc8* layer which is set to 0.01. All other parameters are set to their default values. Next, we prepare a new training set including 40K images from the same four object categories and annotate them with camera view labels. A validation set of size 10K is also constructed. Obtained weights from the first step are loaded to the network and are treated as a promising initialization point for another round of fine tuning over the new data.

We assess the performance of the network for camera view and rotation prediction using the *pool5* representation. As fine tuning with low learning rate slightly changes weights within the network, we are interested to see which order of changes in weights (before fully connected layers) gives the superior performance in our desired task. To hunt what we are looking for, prepared datasets are delivered to the network in reverse order (i.e., camera first, rotation next). We denote the two orderings as follows: 1) *rotation-camera*, and 2) *camera-rotation* for simpler reference. In the evaluation phase, 2,000 samples are randomly selected from four categories, and *pool5* features are extracted. After mean subtraction and dimensionality reduction, 5-fold cross validation accuracies of models are reported in Table. 5.

Counter-intuitively, we find that order of data delivery is very important to the network such that when the network is fed with samples with rotation labels prior to camera labels, it ostensibly performs better in parameter prediction. We also find that when the network is firstly fine tuned on rotation, the second stage (i.e., fine tuning on camera labels) does not impair the weights for rotation prediction. In contrast, when the camera labels are seen first, rotation prediction accuracy is expectedly better than the previous ordering. This boost, however, causes dramatic degradation in camera prediction performance.

Parameter \ Order	1 [rotation-camera]	2 [camera-rotation]
Camera	89.20% (1.47)	77.05% (1.18)
Rotation	93.75%(1.66)	95.30% (1.00)

Table 5. Influence of data delivery order on parameter prediction.

As in the previous experiments, camera view variation is a more ill-structured parameter to predict. When the network sees the camera labels in the second stage, the adapted weights are more biased towards learning this parameter. This bias does also try to keep the pre-seen knowledge for rotation unchanged. We thus conclude that when there is the option for stage-wise training, it would be better to learn parameters following a simple to complex order. This way, the last steps are devoted to manage the difficulties in complex parameters, while imposing less damage to weights adapted for simpler parameters (thus maintaining the structure).

5. Discussion

We challenged the solitary use of uncontrolled natural image datasets in guiding the object recognition progress and introduced a large-scale controlled object dataset of over 20M images with a rich parameter variety. By cutting slices through our dataset, we systematically studied the invariance and generalization properties of CNNs by independently varying the choice of object instances, viewpoints, lighting conditions, or backgrounds between training and test sets. Progressively extending these results on increasingly larger subsets of our dataset may help gain new insights on how the algorithms can be modified to show greater invariance and generalization capabilities.

In summary, we learn that: *i)* the representation learned in *pool5* layer is selective to parameters while *fc7* layer is not, *ii)* the knowledge obtained from some parameters is easier to be transferred to unseen object categories, *iii)* random sampling strategy leads to better generalization since more instance level variations can be captured, *iv)* simple cross application of one dataset to another results in above chance accuracy but does not improve performance, and *v)* it would be advantageous to feed the network with data that has been sorted according to complexities of different dimensions. This can lead to layer-wise training of CNNs for learning different invariances in different layers.

In the future, we will attempt to evaluate the accuracy of recent deep learning architectures on our dataset. In particular, we will consider techniques such as feature embedding and loss regularization [63, 5] and joint prediction of camera parameters and object categories [13, 53].

Acknowledgments: This work was supported by the National Science Foundation (grant numbers CCF-1317433 and CNS-1545089), the Army Research Office (W911NF-12-1-0433), and the Office of Naval Research (N00014-13-1-0563). The authors affirm that the views expressed herein are solely their own, and do not represent the views of the United States government or any agency thereof. We also wish to thank NVIDIA for their generous donation of the GPU used in this study.

References

[1] H. Azizpour, A. Razavian, J. Sullivan, A. Maki, and S. Carlsson. From generic to specific deep representations for visual recognition. In *CVPR Workshops*, pages 36–45, 2015. 5

[2] A. Bakry, M. Elhoseiny, T. El-Gaaly, and A. Elgammal. Digging deep into the layers of cnns: In search of how cnns achieve view invariance. *arXiv:1508.01983*, 2015. 5

[3] P. Bolettieri, A. Esuli, F. Falchi, C. Lucchese, R. Perego, and F. Rabitti. Enabling content-based image retrieval in very large digital libraries. In *Second Workshop on Very Large Digital Libraries (VLDL 2009), 2 October 2009, Corfu, Greece*, pages 43–50, 2009. 2

[4] A. Borji and L. Itti. Human vs. computer in scene and object recognition. In *Computer Vision and Pattern Recognition (CVPR)*, pages 113–120. IEEE, 2014. 2

[5] J. Bromley, J. W. Bentz, L. Bottou, I. Guyon, Y. LeCun, C. Moore, E. Säckinger, and R. Shah. Signature verification using a siamese time delay neural network. *International Journal of Pattern Recognition and Artificial Intelligence*, 7(04):669–688, 1993. 3, 8

[6] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille. Semantic image segmentation with deep convolutional nets and fully connected crfs. *arXiv:1412.7062*, 2014. 1

[7] M. Cimpoi, S. Maji, and A. Vedaldi. Deep filter banks for texture recognition and segmentation. In *CVPR*, pages 3828–3836, 2015. 1

[8] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR 2009*, pages 248–255, 2009. 1, 2, 3

[9] J. J. DiCarlo, D. Zoccolan, and N. C. Rust. How does the brain solve visual object recognition? *Neuron*, 73(3):415–434, 2012. 1

[10] A. Dosovitskiy and T. Brox. Inverting visual representations with convolutional networks. *arXiv:1506.02753*, 2015. 5

[11] B. A. Draper, U. Ahlrichs, and D. Paulus. Adapting object recognition across domains: A demonstration. In *Computer Vision Systems*, pages 256–267. 2001. 2

[12] K. A. Ehinger, B. Hidalgo-Sotelo, A. Torralba, and A. Oliva. Modeling search for people in 900 scenes: A combined source model of eye guidance. *Visual Cognition*, 17:945–978, 2009. 4

[13] M. Elhoseiny, T. El-Gaaly, A. Bakry, and A. Elgammal. Convolutional models for joint object categorization and pose estimation. *arXiv:1511.05175*, 2015. 8

[14] M. Everingham, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman. The pascal visual object classes (voc) challenge. *IJCV*, 88(2):303–338, 2010. 1, 2

[15] L. Fei-Fei, R. Fergus, and P. Perona. Learning generative visual models from few training examples: an incremental bayesian approach tested on 101 object categories. In *CVPR Workshop on Generative-Model Based Vision*, 2004. 2

[16] B. Fernando, A. Habrard, M. Sebban, and T. Tuytelaars. Unsupervised visual domain adaptation using subspace alignment. In *CVPR*, pages 2960–2967, 2013. 2

[17] V. Ferrari, T. Tuytelaars, and L. V. Gool. Object detection with contour segment networks. In *ECCV*, 2006. 2

[18] K. Fukushima. Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. *Biological cybernetics*, 36(4):193–202, 1980. 2

[19] J.-M. Geusebroek, G. J. Burghouts, and A. W. Smeulders. The amsterdam library of object images. *International Journal of Computer Vision*, 61(1):103–112, 2005. 1, 2, 3

[20] R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *CVPR*, pages 580–587, 2014. 1

[21] R. Gopalan, R. Li, and R. Chellappa. Domain adaptation for object recognition: An unsupervised approach. In *ICCV*, pages 999–1006, 2011. 2

[22] G. Griffin, A. Holub, and P. Perona. Caltech-256 object category dataset. 2007. 2, 3

[23] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. *arXiv:1512.03385*, 2015. 1

[24] K. He, X. Zhang, S. Ren, and J. Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *ICCV*, pages 1026–1034, 2015. 2

[25] G. Heitz and D. Koller. Learning spatial context: Using stuff to find things. In *ECCV*, volume 5302, pages 30–43. 2008. 4

[26] D. Held, S. Thrun, and S. Savarese. Deep learning for single-view instance recognition. *arXiv:1507.08286*, 2015. 2

[27] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell. Caffe: Convolutional architecture for fast feature embedding. In *Proceedings of the ACM International Conference on Multimedia*, pages 675–678. ACM, 2014. 5

[28] A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar, and L. Fei-Fei. Large-scale video classification with convolutional neural networks. In *CVPR*, 2014. 1

[29] D. Koubaroulis, J. Matas, J. Kittler, and C. CMP. Evaluating colour-based object recognition algorithms using the soil-47 database. In *Asian Conference on Computer Vision*, volume 2002, page 2, 2002. 1, 2

[30] A. Krizhevsky and G. Hinton. Learning multiple layers of features from tiny images, 2009. 2

[31] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012. 1, 4

[32] S. S. Kruthiventi, K. Ayush, and R. V. Babu. Deepfix: A fully convolutional neural network for predicting human eye fixations. *arXiv:1510.02927*, 2015. 1

[33] K. Lai, L. Bo, X. Ren, and D. Fox. A large-scale hierarchical multi-view rgb-d object dataset. In *ICRA*, 2011. 2

[34] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998. 2, 3

[35] Y. LeCun, F. J. Huang, and L. Bottou. Learning methods for generic object recognition with invariance to pose and lighting. In *CVPR*, volume 2, pages II–97, 2004. 1, 2, 3

[36] J. Z. Leibo, J. Mutch, and T. Poggio. Why The Brain Separates Face Recognition From Object Recognition. In *Advances in Neural Information Processing Systems (NIPS)*, Granada, Spain, 2011. 6

[37] N. Li and J. J. DiCarlo. Unsupervised natural visual experience rapidly reshapes size-invariant object representation in inferior temporal cortex. *Neuron*, 67(6):1062–1075, 2010. 3

[38] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick. Microsoft coco: Common objects in context. In *ECCV*, pages 740–755. 2014. 1, 2

[39] T.-Y. Lin, A. RoyChowdhury, and S. Maji. Bilinear cnn models for fine-grained visual recognition. In *ICCV*, pages 1449–1457, 2015. 1

[40] J. L. Long, N. Zhang, and T. Darrell. Do convnets learn correspondence? In *Advances in Neural Information Processing Systems*, pages 1601–1609, 2014. 5

[41] S. Nene, S. Nayar, and H. Murase. Columbia object image library (coil 100) 1996. *Columbia University*, 1988. 1, 2, 3

[42] A. Opelt, M. Fussenegger, A. Pinz, and P. Auer. Generic object recognition with boosting, 2004. Technical Report TR-EMT-2004-01, EMT, TU Graz, Austria. 1, 2

[43] X. Peng, B. Sun, K. Ali, and K. Saenko. Exploring invariances in deep convolutional neural networks using synthetic images. *CoRR, abs/1412.7122*, 2, 2014. 1, 2

[44] P. J. Phillips, H. Moon, S. A. Rizvi, and P. J. Rauss. The feret evaluation methodology for face-recognition algorithms. *Pattern Analysis and Machine Intelligence, IEEE Trans. on*, 22(10):1090–1104, 2000. 2, 3

[45] N. Pinto, Y. Barhomi, D. D. Cox, and J. J. DiCarlo. Comparing state-of-the-art visual features on invariant object recognition tasks. In *Applications of computer vision (WACV), 2011 IEEE workshop on*, pages 463–470. IEEE, 2011. 1

[46] M. Riesenhuber and T. Poggio. Hierarchical models of object recognition in cortex. *Nature neuroscience*, 2(11):1019–1025, 1999. 2

[47] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, et al. Imagenet large scale visual recognition challenge. *Int. J. of Computer Vision*, 115(3):211–252, 2015. 2

[48] B. C. Russell, A. Torralba, K. P. Murphy, and W. T. Freeman. Labelme: a database and web-based tool for image annotation. *International journal of computer vision*, 77(1-3):157–173, 2008. 1, 2

[49] K. Saenko, B. Kulis, M. Fritz, and T. Darrell. Adapting visual category models to new domains. In *Computer Vision–ECCV 2010*, pages 213–226. Springer, 2010. 2

[50] P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus, and Y. LeCun. Overfeat: Integrated recognition, localization and detection using convolutional networks. *arXiv:1312.6229*, 2013. 1

[51] T. Serre, A. Oliva, and T. Poggio. A feedforward architecture accounts for rapid categorization. *Proceedings of the National Academy of Sciences*, 104(15):6424–6429, 2007. 2

[52] T. Serre, L. Wolf, S. Bileschi, M. Riesenhuber, and T. Poggio. Robust object recognition with cortex-like mechanisms. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 29(3):411–426, 2007. 2

[53] K. Simonyan and A. Zisserman. Two-stream convolutional networks for action recognition in videos. In *NIPS*, pages 568–576, 2014. 1, 8

[54] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv:1409.1556*, 2014. 1

[55] A. Singh, J. Sha, K. S. Narayan, T. Achim, and P. Abbeel. Bigbird: A large-scale 3d database of object instances. In *ICRA*, pages 509–516, 2014. 1, 2

[56] H. Su, S. Maji, E. Kalogerakis, and E. Learned-Miller. Multi-view convolutional neural networks for 3d shape recognition. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 945–953, 2015. 1

[57] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–9, 2015. 1

[58] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf. Deepface: Closing the gap to human-level performance in face verification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1701–1708, 2014. 2

[59] V. S. Tomar and R. C. Rose. Manifold regularized deep neural networks. In *INTERSPEECH*, pages 348–352, 2014. 3

[60] A. Torralba and A. A. Efros. Unbiased look at dataset bias. In *CVPR*, pages 1521–1528. IEEE, 2011. 1, 7

[61] A. Torralba, R. Fergus, and W. T. Freeman. 80 million tiny images: A large data set for nonparametric object and scene recognition. *Pattern Analysis and Machine Intelligence, IEEE Trans. on*, 30(11):1958–1970, 2008. 1, 2

[62] L. Van der Maaten and G. Hinton. Visualizing data using t-sne. *Journal of Machine Learning Research*, 9(2579-2605):85, 2008. 5, 7

[63] J. Weston, F. Ratle, H. Mobahi, and R. Collobert. Deep learning via semi-supervised embedding. In *Neural Networks: Tricks of the Trade*, pages 639–655. Springer, 2012. 3, 8

[64] J. Xiao, J. Hays, K. A. Ehinger, A. Oliva, and A. Torralba. Sun database: Large-scale scene recognition from abbey to zoo. In *Computer vision and pattern recognition (CVPR), 2010 IEEE conference on*, pages 3485–3492, 2010. 1, 2

[65] D. L. Yamins, H. Hong, C. F. Cadieu, E. A. Solomon, D. Seibert, and J. J. DiCarlo. Performance-optimized hierarchical models predict neural responses in higher visual cortex. *Proceedings of the National Academy of Sciences*, 111(23):8619–8624, 2014. 2

[66] S. Yang, P. Luo, C.-C. Loy, and X. Tang. From facial parts responses to face detection: A deep learning approach. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3676–3684, 2015. 1

[67] Y. Yang and T. M. Hospedales. A unified perspective on multi-domain and multi-task learning. *arXiv:1412.7489*, 2014. 2

[68] J. Yosinski, J. Clune, Y. Bengio, and H. Lipson. How transferable are features in deep neural networks? In *NIPS*, pages 3320–3328, 2014. 5

[69] Y. Yuan, L. Mou, and X. Lu. Scene recognition by manifold regularized deep learning architecture. *Neural Networks and Learning Systems, IEEE Trans. on*, 26(10):2222–2233, 2015. 3

[70] M. D. Zeiler and R. Fergus. Visualizing and understanding convolutional networks. In *ECCV*, pages 818–833. 2014. 5